

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **2000078030 A**(43) Date of publication of application: **14.03.00**

(51) Int. Cl.

H03M 13/27
G06F 11/10
(21) Application number: **10243718**(71) Applicant: **DENSO CORP**(22) Date of filing: **28.08.98**
(72) Inventor: **ITO TAKAFUMI**
AONO TAKAYUKI
**(54) INTERLEAVING ADDRESS GENERATOR AND
 INTERLEAVING ADDRESS GENERATION
 METHOD**

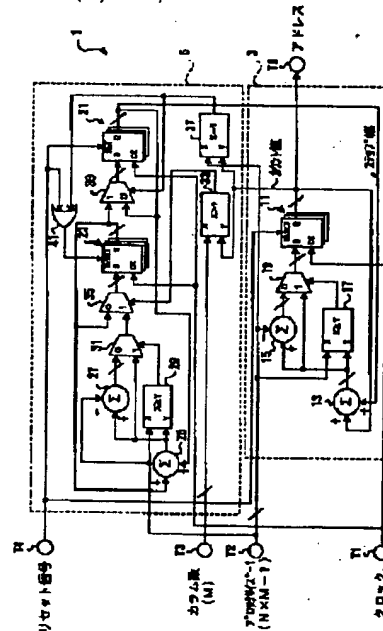
width till then by M whenever the count value returns to '0'.

COPYRIGHT: (C)2000,JPO

(57) Abstract:

PROBLEM TO BE SOLVED: To generate a read/write address to a memory so that a continuous interleaving processing on data of a plurality of blocks is realized with one memory.

SOLUTION: An address generator 1 generating a read/write address to a memory for executing an interleaving on N rows \times M columns where $N \times M$ data are made into one block is provided with an address counter 3 executing count-up from zero at step width which is synchronized with a clock and is inputted, outputting the count value as the address, setting a value obtained by modulo-operating the value by $(N \times M - 1)$ to be the next count value when the count value at the next clock becomes not less than $(N \times M)$, and returning the next count value to '0' when the count value becomes $(N \times M - 1)$ and a step width computing element 5 changing step width to the counter 3 to a value obtained by $(N \times M - 1)$ modulo-operating a value obtained by multiplying step



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-78030

(P2000-78030A)

(43) 公開日 平成12年3月14日 (2000.3.14)

(51) Int.Cl. ⁷	識別記号	F I	テマコード* (参考)
H 0 3 M 13/27		H 0 3 M 13/22	5 B 0 0 1
G 0 6 F 11/10	3 3 0	G 0 6 F 11/10	3 3 0 F 5 J 0 6 5

審査請求 未請求 請求項の数 4 O L (全 17 頁)

(21) 出願番号 特願平10-243718

(22) 出願日 平成10年8月28日 (1998.8.28)

(71) 出願人 000004260

株式会社デンソー

愛知県刈谷市昭和町1丁目1番地

(72) 発明者 伊藤 隆文

愛知県刈谷市昭和町1丁目1番地 株式会
社デンソー内

(72) 発明者 青野 孝之

愛知県刈谷市昭和町1丁目1番地 株式会
社デンソー内

(74) 代理人 100082500

弁理士 足立 勉

Fターム(参考) 5B001 AB02 AC05 AD06 AE04

5J065 AA03 AB01 AC02 AG06 AH02

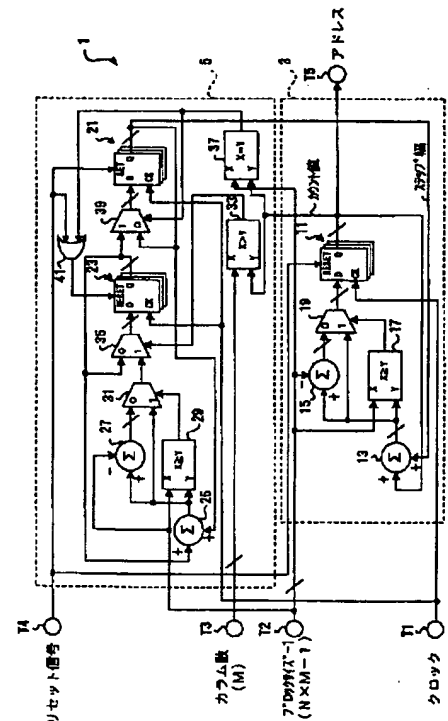
AH04 AH06 AH09 AH17

(54) 【発明の名称】 インターリーブアドレス発生器及びインターリーブアドレス発生方法

(57) 【要約】

【課題】 複数ブロックのデータに対する連続したインターリーブ処理を1つのメモリで実施可能な様にメモリへのリード/ライト用のアドレスを発生させる。

【解決手段】 $N \times M$ 個のデータを1ブロックとした N 行 \times M 列でのインターリーブ処理を行うべくメモリへのリード及びライト用のアドレスを発生するアドレス発生器1は、クロックに同期し且つ入力されるステップ幅で0からカウントアップを行い該カウント値をアドレスとして出力すると共に、次クロックでのカウント値が $(N \times M)$ 以上となる場合には、その値について $(N \times M - 1)$ のモジュロを取った値を次のカウント値とし、更にカウント値が $(N \times M - 1)$ になると次のカウント値を0に戻すアドレスカウンタ3と、該カウンタ3へのステップ幅を、上記カウント値が0に戻る毎に、それまでのステップ幅に M を乗じた値について $(N \times M - 1)$ のモジュロを取った値へと変更するステップ幅演算器5とからなる。



【特許請求の範囲】

【請求項 1】 $N \times M$ 個（但し、 N と M は 2 以上の整数）を 1 ブロックとするデータを順次入力してメモリへ書き込むと共に、そのメモリに書き込んだデータを書き込み時とは異なるアドレスの順序で読み出すことにより、入力したデータを入力時の順序とは異なる順序で出力するインターリーブ処理を行うために用いられ、前記メモリへの読み出し及び書き込み用のアドレスを発生するインターリーブアドレス発生器であって、

クロックに同期し且つ外部より入力されるステップ幅で初期値 0 からカウントアップ動作を行い、そのカウント値を前記メモリへの読み出し及び書き込み用のアドレスとして出力すると共に、次のクロックタイミングでのカウント値が前記 $(N \times M)$ 以上となる場合には、その値について $(N \times M - 1)$ のモジュロを取った値を、次のクロックタイミングでのカウント値とし、更に、今回のカウント値が $(N \times M - 1)$ となった場合には、次のクロックタイミングでのカウント値を初期値 0 に戻すように構成されたアドレスカウント手段と、

前記ステップ幅を演算して前記アドレスカウント手段へ出力する手段であって、現在出力しているステップ幅に前記 M を乗じた値について $(N \times M - 1)$ のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定し、前記アドレスカウント手段のカウント値が初期値 0 に戻ったタイミングに同期して、前記アドレスカウント手段へ出力するステップ幅を前記設定した新たなステップ幅に変更するステップ幅演算手段と、
を備えたことを特徴とするインターリーブアドレス発生器。

【請求項 2】 請求項 1 に記載のインターリーブアドレス発生器において、

前記ステップ幅演算手段は、

前記アドレスカウント手段のカウント値が初期値 0 から $(N \times M - 1)$ となるまでの 1 ブロック分の期間中に、現在出力しているステップ幅を前記 M 回分だけ累積加算すると共に、その累積加算値について $(N \times M - 1)$ のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定するステップ幅設定手段と、

前記アドレスカウント手段のカウント値が $(N \times M - 1)$ になったことを検出する検出手段と、

該検出手段により前記アドレスカウント手段のカウント値が $(N \times M - 1)$ になったことが検出されると、前記ステップ幅設定手段にて設定された新たなステップ幅を、次のクロックタイミングで記憶すると共に、該記憶したステップ幅を前記アドレスカウント手段へ出力するステップ幅出力手段とから構成されていること、
を特徴とするインターリーブアドレス発生器。

【請求項 3】 請求項 1 に記載のインターリーブアドレ

ス発生器において、

前記ステップ幅演算手段は、

現在出力しているステップ幅と前記 M との乗算を行うと共に、その乗算値について $(N \times M - 1)$ のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定するステップ幅設定手段と、

前記アドレスカウント手段のカウント値が $(N \times M - 1)$ になったことを検出する検出手段と、

該検出手段により前記アドレスカウント手段のカウント値が $(N \times M - 1)$ になったことが検出されると、前記ステップ幅設定手段にて設定された新たなステップ幅を、次のクロックタイミングで記憶すると共に、該記憶したステップ幅を前記アドレスカウント手段へ出力するステップ幅出力手段とから構成されていること、
を特徴とするインターリーブアドレス発生器。

【請求項 4】 $N \times M$ 個（但し、 N と M は 2 以上の整数）を 1 ブロックとするデータを順次入力してメモリへ書き込むと共に、そのメモリに書き込んだデータを書き込み時とは異なるアドレスの順序で読み出すことにより、入力したデータを入力時の順序とは異なる順序で出力するインターリーブ処理を行うために用いられ、前記メモリへの読み出し及び書き込み用のアドレスを発生するインターリーブアドレス発生方法であって、

設定されたステップ幅で初期値 0 からカウントアップを行って、そのカウント値を前記メモリへの読み出し及び書き込み用のアドレスとして順次出力すると共に、次のカウント値が前記 $(N \times M)$ 以上となる場合には、その値について $(N \times M - 1)$ のモジュロを取った値を、次のカウント値とし、更に、今回のカウント値が $(N \times M - 1)$ となった場合には、次のカウント値を初期値 0 に戻すようにしたアドレスカウント処理と、

該アドレスカウント処理で用いる前記ステップ幅を設定するステップ幅設定処理とからなり、

前記ステップ幅設定処理では、現在設定しているステップ幅に前記 M を乗じた値について $(N \times M - 1)$ のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として求め、前記アドレスカウント処理によるカウント値が初期値 0 に戻ったタイミングに同期して、前記アドレスカウント処理で用いるステップ幅を前記求めた新たなステップ幅に更新すること、

を特徴とするインターリーブアドレス発生方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、デジタル通信技術に関し、特に、デジタル通信システムにおけるバースト誤りの影響を低減するためのインターリーブ処理に関するものである。

【0002】

【従来の技術】一般に、デジタル通信システムの無線伝送路上で発生するバースト誤りは、誤りデータが集中的に発生するため、無線伝送路で伝送するデータに対してビタビデコーダやリードソロモンデコーダ等による誤り訂正処理を直接適用しても、大きな効果が得られない。

【0003】そこで、従来より、無線伝送路で伝送するデータに対して、誤り訂正処理と共にインターリーブ処理を併用することで、バースト誤りによる集中した誤りデータを分散させてランダム誤りに変換し、これにより、誤り訂正効果の改善を図るようにしている。

【0004】このインターリーブ処理では、送信側でデータの順序を入れ替えて送信するようにしている。そして、受信側で送信側と逆の操作を行うことにより、データを元の順序に戻している。こうすることにより、伝送路でのバースト誤りが分散されてランダムな誤りに変換されるため、誤り訂正効果を高めることができる。

【0005】ここで、こうしたインターリーブ処理において、 $N \times M$ 個（但し、 N と M は2以上の整数）を1ブロックとするデータの順序を入れ替えるために、RAM（ランダムアクセスメモリ）を用いる方法がある。そして、この方法では、RAMに N 行の行アドレスと M 列の列アドレスがあるものと仮定して、その行アドレスの順でRAMに入力データを書き込み、その書き込み終了後に、列アドレスの順でRAMからデータを読み出すことにより、仮想的な行アドレスと列アドレスとの $X-Y$ 変換を行って、入力データを入力時の順序とは異なる順序で出力するようにしている。

【0006】例えば、図7に示すように、こうしたインターリーブ処理を行う従来のインターリーブ回路101は、データ入力端子DINからの入力データを格納すると共に、その格納したデータをデータ出力端子DOUTから出力するRAM103と、クロック入力端子CLKからのクロックに同期して、初期値0から1ずつカウントアップ動作を行う $n+1$ ビットのアドレスカウンタ105、及び、アドレスカウンタ105から出力される $n+1$ ビットのデータ $Q_0 \sim Q_n$ のうちで、最上位ビット Q_n 以外の n ビットデータ $Q_0 \sim Q_{n-1}$ が、 n ビット分の第1入力端子 $a_0 \sim a_{n-1}$ に入力されると共に、アドレスカウンタ105からの上記 n ビットデータ $Q_0 \sim Q_{n-1}$ における下位 m ビット $Q_0 \sim Q_{m-1}$ と上位 $(n-m)$ ビット $Q_m \sim Q_{n-1}$ とを入れ替えた n ビットデータ $Q_m \sim Q_{n-1}$ 、 $Q_0 \sim Q_{m-1}$ が、 n ビット分の第2入力端子 $b_0 \sim b_{n-1}$ に入力されたアドレスセクタ107からなるインターリーブアドレス発生器とを備えている。

【0007】そして、アドレスセクタ107は、アドレスカウンタ105の最上位ビット Q_n が例えば「0」の場合には、第1入力端子 $a_0 \sim a_{n-1}$ に入力されるアドレスカウンタ105からのデータ $Q_0 \sim Q_{n-1}$ を、自己の出力端子 $Y_0 \sim Y_{n-1}$ からRAM103のアドレス

入力端子 $A_0 \sim A_{n-1}$ へ書き込みアドレスとして出力し、逆に、アドレスカウンタ105の最上位ビット Q_n が「1」の場合には、第2入力端子 $b_0 \sim b_{n-1}$ に入力されるアドレスカウンタ105からのデータ $Q_m \sim Q_{n-1}$ 、 $Q_0 \sim Q_{m-1}$ を、自己の出力端子 $Y_0 \sim Y_{n-1}$ からRAM103のアドレス入力端子 $A_0 \sim A_{n-1}$ へ読み出しアドレスとして出力する。

【0008】そして更に、RAM103は、アドレスカウンタ105の最上位ビット Q_n が「0」の場合には、データ入力端子DINからクロックに同期して順次入力されるデータを、アドレス入力端子 $A_0 \sim A_{n-1}$ に入力されるデータにより示されるアドレスに格納し、また、アドレスカウンタ105の最上位ビット Q_n が「1」の場合には、アドレス入力端子 $A_0 \sim A_{n-1}$ に入力されるデータにより示されるアドレスのデータを、クロックに同期してデータ出力端子DOUTから順次出力する。

【0009】このようなインターリーブ回路101では、入力データをRAM103に書き込む場合には、アドレスセクタ107によって、アドレスカウンタ105からの n ビットデータ $Q_0 \sim Q_{n-1}$ が、そのままRAM103のアドレス入力端子 $A_0 \sim A_{n-1}$ へ書き込みアドレスとして出力されるため、RAM103の0番地から順にデータが書き込まれることとなる。

【0010】また、RAM103からデータを読み出す場合には、アドレスセクタ107によって、アドレスカウンタ105からの n ビットデータ $Q_0 \sim Q_{n-1}$ における下位 m ビット $Q_0 \sim Q_{m-1}$ と上位 $(n-m)$ ビット $Q_m \sim Q_{n-1}$ とを入れ替えた n ビットデータ $Q_m \sim Q_{n-1}$ 、 $Q_0 \sim Q_{m-1}$ が、RAM103のアドレス入力端子 $A_0 \sim A_{n-1}$ へ読み出しアドレスとして出力されるため、データの書き込み時に対してRAM103の行アドレスと列アドレスとを入れ替えた順序でデータが読み出されることとなる。

【0011】例えば、4行×4列でのインターリーブ処理を行った場合（即ち、4×4個のデータを1ブロックとしてインターリーブ処理を行った場合）、データ入力端子DINからの入力データは、図8（A）に示すアドレスの順序でRAM103に書き込まれることとなる。そして、1ブロック分の16（=4×4）個のデータが全て書き込まれると、次に、図8（B）に示すアドレスの順序でRAM103からデータが読み出され、その読み出されたデータがデータ出力端子DOUTから順次出力されることとなる。尚、図8（A）における○内の番号は、データを書き込むアドレスの順序を示しており、図8（B）における△内の番号は、データを読み出すアドレスの順序を示している。

【0012】よって、1ブロック分の入力データに対して1番目のものから順にD1, D2, D3, ..., D14, D15, D16というインデックスを付けるとすると、データ出力端子DOUTからは、入力データが、D1, D5

, D9, D13, D2, D6, D10, D14, D3, D7, D11, D15, D4, D8, D12, D16という順に並べ替えられて出力されることとなる。

【0013】

【発明が解決しようとする課題】ところで、上記従来のインターリーブ回路101では、RAM103からのデータ読み出しを終えるまでは、1ブロック分のデータを保持する必要がある。よって、順次入力される複数のブロックのデータを連続してインターリーブ処理するためには、図9に示すように、図7のインターリーブ回路101と同じ構成を有する2つの回路ブロック101A、101Bと、一方の回路ブロック101Bへクロックが供給され始めるのを1ブロック分だけ遅延させる遅延回路109と、他方の回路ブロック101Aにおけるアドレスカウンタ105の最上位ビットQnが「1」の場合には、回路ブロック101Aのデータ出力端子DOUから出力されるデータを出力データとして出力し、逆に、上記最上位ビットQnが「0」の場合には、回路ブロック101Bのデータ出力端子DOUから出力されるデータを出力データとして出力するセクタ111とか

ら、インターリーブ回路を構成し、回路ブロック101A、101Bの各々が、図10に示すようなタイミングで、RAM103にデータを書き込むフェーズと、RAM103からデータを読み出すフェーズとを、交互に受け持つようにする必要がある。

【0014】尚、図10は、図8と同様に、4×4個のデータを1ブロックとしてインターリーブ処理を行った場合を示している。そして、図10において、Qnが「1」の場合に出力されている番号が1～16までの出力データと、番号が33以上の出力データは、回路ブロック101Aによってインターリーブ処理されたものであり、Qnが「0」の場合に出力されている番号が17～32までの出力データは、回路ブロック101Bによってインターリーブ処理されたものである。

【0015】このように、従来の技術では、順次入力される複数のブロックのデータに対してインターリーブ処理を連続して行うためには、アドレスカウンタ105及びアドレスセクタ107からなるインターリーブアドレス発生器と、RAM103とを、2つずつ設ける必要があり、インターリーブ処理を行うための回路を小型化することが困難であった。そして特に、この種のインターリーブ処理は、移動体通信機器等で実施されるため、インターリーブ処理を行う回路は小型であることが要求される。

【0016】そこで、本発明は、複数のブロックのデータに対する連続したインターリーブ処理を、1つのメモリで実施することのできるインターリーブアドレス発生器及びインターリーブアドレス発生方法を提供することを目的としている。

【0017】

【課題を解決するための手段、及び発明の効果】上記目的を達成するためになされた本発明のインターリーブアドレス発生方法は、請求項4に記載のように、N×M個（但し、NとMは2以上の整数）を1ブロックとするデータを順次入力してメモリへ書き込むと共に、そのメモリに書き込んだデータを書き込み時とは異なるアドレスの順序で読み出すことにより、入力したデータを入力時の順序とは異なる順序で出力するインターリーブ処理を行うために用いられ、前記メモリへの読み出し及び書き込み用のアドレスを発生するものである。

【0018】そして、本発明のインターリーブアドレス発生方法は、設定されたステップ幅で初期値0からカウントアップを行って、そのカウント値を前記メモリへの読み出し及び書き込み用のアドレスとして順次出力すると共に、次のカウント値が前記(N×M)以上となる場合には、その値について(N×M-1)のモジュロを取った値を、次のカウント値とし、更に、今回のカウント値が(N×M-1)となった場合には、次のカウント値を初期値0に戻すようにしたアドレスカウンタ処理と、該アドレスカウンタ処理で用いる前記ステップ幅を設定するステップ幅設定処理とからなり、ステップ幅設定処理では、現在設定しているステップ幅に前記Mを乗じた値について(N×M-1)のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として求め、前記アドレスカウンタ処理によるカウント値が初期値0に戻ったタイミングに同期して、前記アドレスカウンタ処理で用いるステップ幅を前記求めた新たなステップ幅に更新することを特徴としている。

【0019】また、このような本発明のインターリーブアドレス発生方法は、請求項1に記載のインターリーブアドレス発生器によって実施することができる。即ち、請求項1に記載のインターリーブアドレス発生回路は、N×M個（但し、NとMは2以上の整数）を1ブロックとするデータを順次入力してメモリへ書き込むと共に、そのメモリに書き込んだデータを書き込み時とは異なるアドレスの順序で読み出すことにより、入力したデータを入力時の順序とは異なる順序で出力するインターリーブ処理を行うために用いられ、前記メモリへの読み出し及び書き込み用のアドレスを発生するものであり、アドレスカウンタ手段と、ステップ幅演算手段とを備えている。

【0020】そして、アドレスカウンタ手段は、クロックに同期し且つ外部より入力されるステップ幅で初期値0からカウントアップ動作を行い、そのカウント値を前記メモリへの読み出し及び書き込み用のアドレスとして出力すると共に、次のクロックタイミングでのカウント値が前記(N×M)以上となる場合には、その値について(N×M-1)のモジュロを取った値を、次のクロックタイミングでのカウント値とし、更に、今回のカウント値が(N×M-1)となった場合には、次のクロック

タイミングでのカウント値を初期値 0 に戻すように構成されている。

【0021】そして更に、ステップ幅演算手段は、アドレスカウント手段がカウントアップ動作に用いるステップ幅を演算して前記アドレスカウント手段へ出力するのであるが、特に、現在出力しているステップ幅に前記 M を乗じた値について $(N \times M - 1)$ のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定し、前記アドレスカウント手段のカウント値が初期値 0 に戻ったタイミングに同期して、前記アドレスカウント手段へ出力するステップ幅を前記設定した新たなステップ幅に変更する。

【0022】尚、ステップ幅とは、メモリへのアドレスとなるカウント値をカウントアップする際の、1 回毎の増加分の値を意味している。また、ある値 Z について $(N \times M - 1)$ のモジュロを取った値とは、ある値 Z を $(N \times M - 1)$ で割った場合の余りの値を意味している。

【0023】このような本発明のインターリーブアドレス発生方法及びインターリーブアドレス発生器において、アドレスカウント処理或いはアドレスカウント手段によって出力されるアドレスのステップ幅は、その出力されるアドレスが $(N \times M - 1)$ から初期値 0 に戻る毎であって、入力データの 1 ブロック分毎に変更されることとなる。

【0024】よって、本発明のインターリーブアドレス発生方法及びインターリーブアドレス発生器によれば、1 番目の（即ち、最初の）ブロックのデータを入力する期間においては、アドレスカウント処理或いはアドレスカウント手段によって出力されるメモリの各アドレスに入力データを順次書き込み、2 番目以降の各ブロックのデータを入力する期間においては、アドレスカウント処理或いはアドレスカウント手段によって出力されるメモリの各アドレスからデータを読み出すと共に、その読み出したアドレスに現在の入力データを上書きすることにより、メモリからのデータの読み出しが、そのデータを書き込んだ時とは異なるステップ幅でアドレッシングされることとなる。このため、メモリとインターリーブアドレス発生器とを 1 組備えるだけで、複数のブロックのデータに対する連続したインターリーブ処理が可能となる。

【0025】例えば、ステップ幅の初期値を 1 とすると共に、行数（ロー数） $N = 4$ 及び列数（カラム数） $M = 3$ のブロックサイズでインターリーブ処理を行う場合（即ち、 4×3 個のデータを 1 ブロックとした 4 行 \times 3 列でのインターリーブ処理を行う場合）について、具体的に説明する。

【0026】この場合には、図 4 の上から 5 段目及び最下段に示すように、1 番目のブロックのデータ（D1-1 ～ D1-12）を入力する期間においては、アドレスカウ

ント処理或いはアドレスカウント手段から、0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 の順にアドレスが出力されることとなる。

【0027】そして、2 番目のブロックのデータ（D2-1 ～ D2-12）を入力する期間においては、ステップ幅設定処理或いはステップ幅演算手段によって設定或いは出力されるステップ幅が、1 番目のブロックの入力期間におけるステップ幅（= 1）に M （= 3）を乗じた値（= 3）について 11（= $N \times M - 1$ ）のモジュロを取った値（= 3）となる。よって、アドレスカウント処理或いはアドレスカウント手段からは、0, 3, 6, 9, 1, 4, 7, 10, 2, 5, 8, 11 の順にアドレスが出力されることとなる。尚、出力されるアドレスが 9 の次に 1 へと変化しているのは、9 に現在のステップ幅である 3 を加算した値（= 12）が、1 ブロックのデータ数である 12（= $N \times M$ ）以上となるため、この場合には、12 について 11（= $N \times M - 1$ ）のモジュロを取った値である 1 が、次のカウント値になるからである。また同様に、出力されるアドレスが 10 の次に 2 へと変化しているのも、10 に現在のステップ幅である 3 を加算した値（= 13）が 12（= $N \times M$ ）以上となるため、この場合には、13 について 11（= $N \times M - 1$ ）のモジュロを取った値である 2 が、次のカウント値になるからである。

【0028】そして更に、3 番目のブロックのデータ（D3-1 ～ D3-12）を入力する期間においては、ステップ幅設定処理或いはステップ幅演算手段によって設定或いは出力されるステップ幅が、2 番目のブロックの入力期間におけるステップ幅（= 3）に M （= 3）を乗じた値（= 9）について 11（= $N \times M - 1$ ）のモジュロを取った値（= 9）となる。よって、アドレスカウント処理或いはアドレスカウント手段からは、0, 9, 7, 5, 3, 1, 10, 8, 6, 4, 2, 11 の順にアドレスが出力されることとなる。尚、出力されるアドレスが 9 の次に 7 へと変化しているのは、9 に現在のステップ幅である 9 を加算した値（= 18）が、1 ブロックのデータ数である 12（= $N \times M$ ）以上となるため、この場合には、18 について 11（= $N \times M - 1$ ）のモジュロを取った値である 7 が、次のカウント値になるからである。そして、このことは、出力されるアドレスが 7 の次に 5 へと変化する場合や、5 の次に 3 へと変化する場合等についても同様である。

【0029】また、図 4 には示されていないが、4 番目のブロックのデータを入力する期間においては、ステップ幅設定処理或いはステップ幅演算手段によって設定或いは出力されるステップ幅が、3 番目のブロックの入力期間におけるステップ幅（= 9）に M （= 3）を乗じた値（= 27）について 11（= $N \times M - 1$ ）のモジュロを取った値（= 5）となる。よって、アドレスカウント処理或いはアドレスカウント手段からは、0, 5, 1

0, 4, 9, 3, 8, 2, 7, 1, 6, 11の順にアドレスが出力されることとなる。尚、出力されるアドレスが10の次に4へと変化しているのは、10に現在のステップ幅である5を加算した値(=15)が、1ブロックのデータ数である12(=N×M)以上となるため、この場合には、15について11(=N×M-1)のモジュロを取った値である4が、次のカウント値になるからである。そして、このことは、出力されるアドレスが9の次に3へと変化する場合や、8の次に2へと変化する場合等についても同様である。

【0030】よって、1番目のブロックのデータを、アドレスカウント処理或いはアドレスカウント手段によって出力されるメモリの各アドレスに順次書き込めば、その1番目のブロックの各データは、メモリの0から11までのアドレスへ、図1(A)の○内に示す番号の順序で書き込まれることとなる。

【0031】そして、2番目のブロックのデータを入力する期間において、アドレスカウント処理或いはアドレスカウント手段によって出力されるメモリの各アドレスからデータを読み出すと共に、その読み出したアドレスに現在の入力データを上書きすれば、メモリに既に書き込まれている1番目のブロックの各データは、0から11までのアドレスから、図1(B)の△内に示す番号の順序で読み出され、また、2番目のブロックの各データは、メモリの0から11までのアドレスへ、図1(C)の○内に示す番号の順序(即ち、図1(B)の△内に示す番号と同じ順序)で書き込まれることとなる。

【0032】そして更に、3番目のブロックのデータを入力する期間において、アドレスカウント処理或いはアドレスカウント手段によって出力されるメモリの各アドレスからデータを読み出すと共に、その読み出したアドレスに現在の入力データを上書きすれば、メモリに既に書き込まれている2番目のブロックの各データは、0から11までのアドレスから、図1(D)の△内に示す番号の順序で読み出され、また、3番目のブロックの各データは、図1(B)と図1(C)の関係と同様に、メモリの0から11までのアドレスへ図1(D)の△内に示す番号の順序で書き込まれることとなる。

【0033】このため、1ブロック分の12個の入力データに対して1番目のものから順にD1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12というインデックスを付けるとすると、図1(A)と図1(B)の関係及び図1(C)と図1(D)の関係からも分かるように、順次入力される各ブロックの入力データは、メモリから読み出されることにより、夫々、D1, D4, D7, D10, D2, D5, D8, D11, D3, D6, D9, D12という順序に並べ替えられることとなる。

【0034】つまり、メモリへのアドレスのステップ幅を、ステップ幅設定処理或いはステップ幅演算手段によ

ってデータの1ブロック分毎に変更していくことにより、1つのメモリだけで連続したインターリーブ処理が可能となるのである。尚、ステップ幅の初期値は、1に限るものではなく、1以上の整数であれば良い。また、インターリーブ処理に用いるメモリの先頭アドレスが0番地でない場合には、アドレスカウント処理或いはアドレスカウント手段によって出力されるアドレスに上記先頭のアドレス値を加算し、その加算後のアドレスをメモリのアドレス入力端子へ供給するようにすれば良い。

10 【0035】ところで、請求項1に記載のインターリーブアドレス発生器において、ステップ幅演算手段は、請求項2に記載のように、アドレスカウント手段のカウント値が初期値0から(N×M-1)となるまでの1ブロック分の期間中に、現在出力しているステップ幅を前記M回分だけ累積加算すると共に、その累積加算値について(N×M-1)のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定するステップ幅設定手段と、アドレスカウント手段のカウント値が(N×M-1)になったことを検出する検出手段と、該検出手段によりアドレスカウント手段のカウント値が(N×M-1)になったことが検出されると、前記ステップ幅設定手段にて設定された新たなステップ幅を、次のクロックタイミングで記憶すると共に、該記憶したステップ幅をアドレスカウント手段へ出力するステップ幅出力手段と、から構成することができる。

20 【0036】つまり、請求項2に記載のインターリーブアドレス発生器では、現在出力しているステップ幅に前記Mを乗じた値を、現在のステップ幅を前記M回分だけ累積加算することによって求めている。また、請求項1に記載のインターリーブアドレス発生器において、ステップ幅演算手段は、請求項3に記載のように、現在出力しているステップ幅と前記Mとの乗算を行うと共に、その乗算値について(N×M-1)のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定するステップ幅設定手段と、アドレスカウント手段のカウント値が(N×M-1)になったことを検出する検出手段と、該検出手段によりアドレスカウント手段のカウント値が(N×M-1)になったことが検出されると、前記ステップ幅設定手段にて設定された新たなステップ幅を、次のクロックタイミングで記憶すると共に、該記憶したステップ幅をアドレスカウント手段へ出力するステップ幅出力手段と、から構成することもできる。

40 【0037】つまり、請求項3に記載のインターリーブアドレス発生器では、現在出力しているステップ幅に前記Mを乗じた値を、現在のステップ幅と前記Mとを乗算することによって求めている。但し、請求項2に記載のように、現在出力しているステップ幅をM回累積加算するように構成する方が、乗算を直接行う必要がないた

め、回路構成を簡素化できるという点で有利である。

【0038】

【発明の実施の形態】以下、本発明の一実施形態について、図面を用いて説明する。まず図2は、第1実施形態のインターリーブアドレス発生器1の構成を表すブロック図であり、図3は、そのインターリーブアドレス発生器1の詳細な回路図である。

【0039】尚、本第1実施形態のインターリーブアドレス発生器1は、 $N \times M$ 個（但し、 N と M は2以上の整数）のデータを1ブロックとした N 行 M 列でのインターリーブ処理を行うために、図示しないメモリ（具体的にはRAM）への読み出し及び書き込み用のアドレスを発生するものである。また、以下の説明において、上記 M を、列数を意味する「カラム数」といい、1ブロックのデータ数である上記 $(N \times M)$ を、「ブロックサイズ」という。

【0040】図2に示すように、本第1実施形態のインターリーブアドレス発生器1は、クロック入力端子T1からのクロックに同期し且つ外部より入力されるステップ幅で初期値0からカウントアップ動作を行い、そのカウント値をメモリへの読み出し及び書き込み用のアドレスとして出力するアドレスカウンタ3と、そのアドレスカウンタ3から出力されるアドレス、ブロックサイズ入力端子T2から入力されるブロックサイズ-1（ $=N \times M - 1$ ）を表すデータ、及びカラム数入力端子T3から入力されるカラム数（ $=M$ ）を表すデータに基づき、クロック入力端子T1からのクロックに同期して、アドレスカウンタ3でのステップ幅を演算し出力するステップ幅演算器5とから構成されている。

【0041】尚、アドレスカウンタ3とステップ幅演算器5には、リセット信号入力端子T4を介して、ハイアクティブのリセット信号が入力されるようになっている。また、アドレスカウンタ3にも、ブロックサイズ入力端子T2からのブロックサイズ-1（ $=N \times M - 1$ ）を表すデータが入力されている。また更に、アドレスカウンタ3からメモリへは、アドレス出力端子T5を介してアドレスが出力されるようになっている。

【0042】次に、図3に示すように、アドレスカウンタ3は、当該アドレスカウンタ3のカウント値をクロック入力端子T1からのクロックに同期して記憶し、その記憶したカウント値をメモリへのアドレスとしてアドレス出力端子T5に出力する、入力バス幅に対応した数のクロック同期リセット機能付きフリップフロップからなるフリップフロップ群11（以下、F/F群11という）と、F/F群11の出力値（即ち、当該アドレスカウンタ3の現在のカウント値であって、現在出力しているメモリへのアドレス）とステップ幅演算器5から入力されるステップ幅とを加算する加算器13と、その加算器13の演算結果から、ブロックサイズ入力端子T2より入力されるブロックサイズ-1を表すデータ（以下単

に、ブロックサイズ-1という）を減算する減算器15と、加算器13の演算結果とブロックサイズ入力端子T2からのブロックサイズ-1とを大小比較して、加算器13の演算結果がブロックサイズ-1（ $=N \times M - 1$ ）以下である場合に、ハイレベルのセレクト信号を出力し、そうでなければ（つまり、加算器13の演算結果がブロックサイズ（ $=N \times M$ ）以上である場合に）、ロウレベルのセレクト信号を出力するコンパレータ17と、コンパレータ17からのセレクト信号がハイレベルの場合に、加算器13の演算結果をF/F群11のデータ端子（D）に入力させ、コンパレータ17からのセレクト信号がロウレベルの場合に、減算器15の演算結果をF/F群11のデータ端子（D）に入力させるセクタ19と、から構成されている。

【0043】尚、F/F群11は、クロックの立ち上がり同期して動作し、リセット信号入力端子T4から自己のリセット端子（RESET）へ入力されるリセット信号がロウレベルである通常時には、セクタ19から入力バスを介して自己のデータ端子（D）に入力されるデータ値を、クロックの立ち上がりタイミングで記憶（ラッチ）して出力端子（Q）からアドレス出力端子T5へ出力する。また、F/F群11は、自己の出力値がブロックサイズ-1（ $=N \times M - 1$ ）になると、次のクロックの立ち上がりタイミングでは、セクタ19の出力に拘わらず、その出力値が初期値である0へ戻るように構成されている。

【0044】このように構成されたアドレスカウンタ3では、リセット信号入力端子T4からハイレベルのリセット信号が入力されると、F/F群11が、クロック入力端子T1からのクロックに同期してリセットされ、その結果、F/F群11に記憶されるカウント値であって、アドレス出力端子T5からメモリへ出力されるアドレスは、全ビットが0の初期値0となる（図4参照）。

【0045】そして、リセット信号入力端子T4からのリセット信号がロウレベルになって、リセットが解除されると、そのリセット解除後の次のクロックより、F/F群11を中心としてメモリへのアドレスのカウントアップ動作が行われる。このアドレスカウンタ3のカウントアップ動作としては、まず、加算器13により、F/F群11の現在の出力値とステップ幅演算器5から入力されるステップ幅とを加算され、その加算器13の演算結果とブロックサイズ-1（ $=N \times M - 1$ ）の値とが、コンパレータ17によって大小比較される。

【0046】そして、加算器13の演算結果がブロックサイズ-1（ $=N \times M - 1$ ）以下である場合には、コンパレータ17からのセレクト信号に応じて動作するセクタ19により、加算器13の演算結果がそのままF/F群11のデータ端子（D）に入力され、その結果、F/F群11は、次のクロックの立ち上がりタイミングにて、加算器13の演算結果を当該アドレスカウンタ3の

カウント値として記憶すると共に、その記憶したカウント値を自己の出力端子 (Q) からアドレス出力端子 T5 へ、メモリへのアドレスとして出力することとなる。

【0047】また逆に、加算器 13 の演算結果がブロックサイズ ($=N \times M$) 以上である場合には、上記セレクタ 19 により、減算器 15 の演算結果であって、加算器 13 の演算結果からブロックサイズ - 1 ($=N \times M - 1$) を減じた値が、F/F 群 11 のデータ端子 (D) に入力され、その結果、F/F 群 11 は、次のクロックの立ち上がりタイミングにて、減算器 15 の演算結果を当該アドレスカウンタ 3 のカウント値として記憶すると共に、その記憶したカウント値を自己の出力端子 (Q) からアドレス出力端子 T5 へ、メモリへのアドレスとして出力することとなる。

【0048】よって、アドレスカウンタ 3 では、F/F 群 11 及び加算器 13 の作用により、クロックに同期し且つステップ幅演算器 5 より入力されるステップ幅で初期値 0 からカウントアップ動作が行われて、そのカウント値がメモリへのアドレスとして出力されることとなるが、特に、次のクロックタイミングでのカウント値がブロックサイズ ($=N \times M$) 以上となる場合には、減算器 15、コンパレータ 17、及びセレクタ 19 の作用により、そのブロックサイズ以上の値についてブロックサイズ - 1 ($=N \times M - 1$) のモジュロを取った値が、次のクロックタイミングでのカウント値となる。そして更に、今回のクロックタイミングでのカウント値がブロックサイズ - 1 ($=N \times M - 1$) になると、前述した F/F 群 11 の作用により、次のクロックタイミングでのカウント値が初期値 0 に戻る事となる。

【0049】一方、図 3 に示すように、ステップ幅演算器 5 は、アドレスカウンタ 3 が前述のカウントアップ動作に用いるステップ幅をクロック入力端子 T1 からのクロックに同期して記憶し、その記憶したステップ幅をアドレスカウンタ 3 (詳しくは、アドレスカウンタ 3 の加算器 13) へ出力する、1 ビット目がクロック同期プリセット機能付きのフリップフロップ、それ以外がクロック同期リセット機能付きのフリップフロップからなるフリップフロップ群 21 (以下、F/F 群 21 という) と、F/F 群 21 からアドレスカウンタ 3 へ出力すべきステップ幅を演算する過程で生じる演算値を、クロック入力端子 T1 からのクロックに同期して記憶する、入力バス幅に対応した数のクロック同期リセット機能付きフリップフロップからなるフリップフロップ群 23 (以下、F/F 群 23 という) と、を備えている。

【0050】尚、F/F 群 21、23 は、クロックの立ち上がりに同期して動作し、自己のリセット端子 (RESET) 或いはプリセット端子 (SET) に入力される信号がロウレベルである通常時には、入力バスを介して自己のデータ端子 (D) に入力されるデータ値を、クロックの立ち上がりタイミングで記憶 (ラッチ) して出力

端子 (Q) から出力する。また、ステップ幅演算器 5 は、F/F 群 21 の出力値 (即ち、F/F 群 21 からアドレスカウンタ 3 へ現在出力しているステップ幅) と F/F 群 23 の出力値とを加算する加算器 25 と、その加算器 25 の演算結果から、ブロックサイズ入力端子 T2 より入力されるブロックサイズ - 1 ($=N \times M - 1$) を減算する減算器 27 と、加算器 25 の演算結果とブロックサイズ入力端子 T2 からのブロックサイズ - 1 とを大小比較して、加算器 25 の演算結果がブロックサイズ - 1 ($=N \times M - 1$) 以下である場合に、ハイレベルのセレクト信号を出力し、そうでなければ (つまり、加算器 25 の演算結果がブロックサイズ ($=N \times M$) 以上である場合に)、ロウレベルのセレクト信号を出力するコンパレータ 29 と、コンパレータ 29 からのセレクト信号がハイレベルの場合に、加算器 25 の演算結果を選択して出力し、コンパレータ 29 からのセレクト信号がロウレベルの場合に、減算器 27 の演算結果を選択して出力するセレクタ 31 と、アドレスカウンタ 3 のカウント値 (即ち、アドレスカウンタ 3 の F/F 群 11 からメモリへ出力されているアドレス) とカラム数入力端子 T3 から入力されるカラム数を表すデータ (以下単に、カラム数という) とを大小比較して、アドレスカウンタ 3 のカウント値がカラム数 ($=M$) よりも小さい場合に、ハイレベルのセレクト信号を出力し、そうでなければ (つまり、アドレスカウンタ 3 のカウント値がカラム数 ($=M$) 以上である場合に)、ロウレベルのセレクト信号を出力するコンパレータ 33 と、コンパレータ 33 からのセレクト信号がハイレベルの場合に、セレクタ 31 の出力を F/F 群 23 のデータ端子 (D) に入力させ、コンパレータ 33 からのセレクト信号がロウレベルの場合に、F/F 群 23 の出力を同 F/F 群 23 のデータ端子 (D) に入力させるセレクタ 35 と、を備えている。

【0051】そして更に、ステップ幅演算器 5 は、アドレスカウンタ 3 のカウント値とブロックサイズ入力端子 T2 からのブロックサイズ - 1 とを比較して、アドレスカウンタ 3 のカウント値がブロックサイズ - 1 ($=N \times M - 1$) と等しい場合に、ハイレベルのセレクト信号を出力し、そうでなければ (つまり、アドレスカウンタ 3 のカウント値がブロックサイズ - 1 ($=N \times M - 1$) と等しくない場合に)、ロウレベルのセレクト信号を出力するコンパレータ 37 と、コンパレータ 37 からのセレクト信号がハイレベルの場合に、F/F 群 23 の出力を F/F 群 21 のデータ端子 (D) に入力させ、コンパレータ 37 からのセレクト信号がロウレベルの場合に、F/F 群 21 の出力を同 F/F 群 21 のデータ端子 (D) に入力させるセレクタ 39 と、リセット信号入力端子 T4 からのリセット信号とコンパレータ 37 からのセレクト信号との論理和信号を、F/F 群 23 のリセット端子 (RESET) に入力させる論理和回路 41 と、を備えている。

【0052】次に、上記のように構成されたステップ幅演算器5の動作について説明する。まず、リセット信号入力端子T4からハイレベルのリセット信号が入力されると、F/F群23では、その各フリップフロップのリセット端子(RESET)に入力される信号が、論理和回路41を介してハイレベルとなり、また、F/F群21では、1ビット目のフリップフロップのプリセット端子(SET)と、それ以外のビットのフリップフロップのリセット端子(RESET)とに夫々入力される信号がハイレベルとなる。

【0053】このため、リセット信号入力端子T4からハイレベルのリセット信号が入力されると、クロック入力端子T1からのクロックに同期して、F/F群23の出力値は全ビットが0となる。また、F/F群21の出力値は、1ビット目だけが1で、それ以外のビットが0となり、この結果、アドレスカウンタ3へ出力されるステップ幅は初期値としての1になる(図4参照)。

【0054】その後、リセット信号入力端子T4からのリセット信号がロウレベルになって、リセットが解除されると、F/F群23は、コンパレータ33からのセレクト信号に応じて動作するセクタ35の出力を、クロックに同期してラッチし出力する。

【0055】ここで、セクタ35は、コンパレータ33からのセレクト信号がハイレベルの場合(即ち、アドレスカウンタ3のカウント値であって、アドレス出力端子T5から出力されているアドレスが、カラム数(=M)よりも小さい場合)には、セクタ31の出力をF/F群23のデータ端子(D)に入力させる。

【0056】そして、セクタ31は、コンパレータ29からのセレクト信号に従い、加算器25によってF/F群21の出力値とF/F群23の出力値とが加算された演算結果が、ブロックサイズ入力端子T2からのブロックサイズ-1(=N×M-1)以下である場合には、その加算器25の演算結果をそのままセクタ35へ出力し、逆に、加算器25の演算結果がブロック数(=N×M)以上である場合には、減算器27によって加算器25の演算結果からブロックサイズ-1(=N×M-1)が減算された演算結果を、セクタ35へ出力する。

【0057】また、セクタ35は、コンパレータ33からのセレクト信号がロウレベルの場合(即ち、アドレスカウンタ3の現在のカウント値であって、アドレス出力端子T5から出力されているアドレスが、カラム数(=M)以上である場合)には、F/F群23の出力をF/F群23のデータ端子(D)に入力させる。

【0058】よって、F/F群23は、アドレスカウンタ3のカウント値がカラム数(=M)よりも小さい場合には、現在の自己の出力値とF/F群23の出力値とを加算した値についてブロックサイズ-1(=N×M-1)のモジュロを取った値を、次のクロックタイミング

でラッチし、また、アドレスカウンタ3のカウント値がカラム数(=M)以上である場合には、自己の出力値を更新せずに保持することとなる。

【0059】この結果、F/F群23は、加算器25、減算器27、コンパレータ29、33、及びセクタ31、35と共同して動作することにより、アドレスカウンタ3のカウント値が初期値0からブロックサイズ-1(=N×M-1)となるまでの1ブロック分の期間中に、F/F群21から出力されているステップ幅をカラム数M回分だけ累積加算すると共に、その累積加算値についてブロックサイズ-1(=N×M-1)のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として記憶することとなる。そして、この動作により、アドレスカウンタ3へ現在出力しているステップ幅にカラム数(=M)を乗じた値についてブロックサイズ-1(=N×M-1)のモジュロを取った値が、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定される。

【0060】また、アドレスカウンタ3のカウント値がブロックサイズ-1(=N×M-1)になると、コンパレータ37が、ブロックの最終データの入力タイミングであると判断してハイレベルのセレクト信号を出力する。そして、そのコンパレータ37からのハイレベルのセレクト信号は、論理和回路41を介してF/F群23のリセット端子(RESET)に入力される。よって、F/F群23は、次のクロックタイミング(即ち、アドレスカウンタ3のカウント値が初期値0に戻るタイミングであって、次のブロックの最初のデータを入力するタイミング)でリセットされ、その出力値が0に戻るものとなる。つまり、コンパレータ37から出力されるハイレベルのセレクト信号は、1ブロックの終了を意味するブロック終了信号となっている。

【0061】一方、当該ステップ幅演算器5において、リセット信号入力端子T4からのリセット信号がハイレベルからロウレベルになってリセットが解除されると、F/F群21は、コンパレータ37からのセレクト信号に応じて動作するセクタ39の出力を、クロックに同期してラッチし、ステップ幅として出力する。

【0062】ここで、セクタ39は、コンパレータ37からのセレクト信号がロウレベルの場合(即ち、アドレスカウンタ3のカウント値がブロックサイズ-1(=N×M-1)でない場合)には、F/F群21の出力をF/F群21のデータ端子(D)に入力させ、また、コンパレータ37からのセレクト信号がハイレベルの場合(即ち、アドレスカウンタ3のカウント値がブロックサイズ-1(=N×M-1)である場合)に、F/F群23の出力をF/F群21のデータ端子(D)に入力させる。

【0063】よって、F/F群21は、アドレスカウンタ3のカウント値がブロックサイズ-1(=N×M-

10

20

30

40

50

1) になると、次のクロックタイミング（即ち、アドレスカウンタ 3 のカウント値が初期値 0 に戻るタイミング）で、その直前に F/F 群 2 3 に記憶されている次のブロックに対する新たなステップ幅をラッチし、そのラッチしたステップ幅を、アドレスカウンタ 3 のカウント値が次にブロックサイズ - 1（ $=N \times M - 1$ ）から初期値 0 へ戻るまでの 1 ブロック分の期間中、アドレスカウンタ 3 へ出力することとなる。

【0064】つまり、F/F 群 2 1 は、セクタ 3 9 と共同して動作することにより、コンパレータ 3 7 によってアドレスカウンタ 3 のカウント値がブロックサイズ - 1（ $=N \times M - 1$ ）になったことが検出されると、F/F 群 2 3 にラッチされている新たなステップ幅を次のクロックタイミングで記憶すると共に、その記憶した新たなステップ幅をアドレスカウンタ 3 へ出力することとなる。そして、この動作により、アドレスカウンタ 3 のカウント値が初期値 0 に戻ったタイミングに同期して、そのアドレスカウンタ 3 へ出力されるステップ幅が新たなステップ幅に変更される。

【0065】次に、図 4 は、上述したインターリーブアドレス発生器 1 の動作を、行数（ロー数） $N=4$ 及び列数（カラム数） $M=3$ のブロックサイズでインターリーブ処理を行う場合を例に挙げて表すタイムチャートである。尚、本実施形態のインターリーブアドレス発生器 1 が用いられる場合には、クロック入力端子 T1 へのクロックに同期して、メモリへデータが順次入力される。

【0066】図 4 に示すように、この例の場合には、1 番目のブロックのデータ（D1-1 ~ D1-12）を入力する期間（即ち、リセット信号がハイレベルからロウレベルに変化する直前から、データの 1 ブロック分に相当するクロックの 12 周期分の期間）において、アドレスカウンタ 3 から、0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 の順にアドレスが出力されることとなる。これは、ステップ幅演算器 5 の F/F 群 2 1 から出力されるステップ幅の初期値が、1 であるためである。

【0067】また、この期間においては、ステップ幅演算器 5 の F/F 群 2 3、加算器 2 5、減算器 2 7、コンパレータ 2 9、3 3、及びセクタ 3 1、3 5 からなる回路部分の作用により、アドレスカウンタ 3 からのアドレスがカラム数 $M (=3)$ よりも小さい 0, 1, 2 の各場合に、F/F 群 2 1 から現在出力されているステップ幅（ $=1$ ）が累積加算されると共に、その累積加算値についてブロックサイズ - 1（ $=11$ ）のモジュロを取った値（ $=3$ ）が、次のブロックに対する新たなステップ幅として F/F 群 2 3 に記憶される。尚、この場合には、ステップ幅を 3 回累積加算してもブロックサイズ（ $=12$ ）以上とならないため、F/F 群 2 3 に記憶される値は、 $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ と変化する。

【0068】そして、アドレスカウンタ 3 から出力されるアドレスが 11 から 0 に戻ると、そのタイミングで、

F/F 群 2 3 に記憶されている新たなステップ幅（ $=3$ ）が F/F 群 2 1 にラッチされると共に、F/F 群 2 3 の出力値が 0 に戻り、F/F 群 2 1 にラッチされた新たなステップ幅（ $=3$ ）が、2 番目のブロックのデータ（D2-1 ~ D2-12）を入力する期間におけるステップ幅として、アドレスカウンタ 3 へ出力されることとなる。

【0069】このため、2 番目のブロックのデータ（D2-1 ~ D2-12）を入力する期間において、アドレスカウンタ 3 からは、0, 3, 6, 9, 1, 4, 7, 10, 2, 5, 8, 11 の順にアドレスが出力されることとなる。尚、この期間において、出力されるアドレスが 9 の次に 1 へと変化しているのは、9 に現在のステップ幅である 3 を加算した値（ $=12$ ）が、ブロックサイズ（ $=12$ ）以上となるため、この場合には、減算器 1 5 の演算結果であって、12 についてブロックサイズ - 1（ $=11$ ）のモジュロを取った値である 1 が、次のカウント値として F/F 群 1 1 にラッチされるからである。そして、このことは、出力されるアドレスが 10 の次に 2 へと変化する場合についても同様である。

【0070】また、この期間においては、ステップ幅演算器 5 の F/F 群 2 3、加算器 2 5、減算器 2 7、コンパレータ 2 9、3 3、及びセクタ 3 1、3 5 からなる回路部分の作用により、アドレスカウンタ 3 からのアドレスがカラム数 $M (=3)$ よりも小さい 0, 1, 2 の各場合に、F/F 群 2 1 から現在出力されているステップ幅（ $=3$ ）が累積加算されると共に、その累積加算値についてブロックサイズ - 1（ $=11$ ）のモジュロを取った値（ $=9$ ）が、次のブロックに対する新たなステップ幅として F/F 群 2 3 に記憶される。尚、この場合には、ステップ幅を 3 回累積加算してもブロックサイズ（ $=12$ ）以上とならないため、F/F 群 2 3 に記憶される値は、 $0 \rightarrow 3 \rightarrow 6 \rightarrow 9$ と変化する。

【0071】そして、アドレスカウンタ 3 から出力されるアドレスが 11 から 0 に戻ると、そのタイミングで、F/F 群 2 3 に記憶されている新たなステップ幅（ $=9$ ）が F/F 群 2 1 にラッチされると共に、F/F 群 2 3 の出力値が 0 に戻り、F/F 群 2 1 にラッチされた新たなステップ幅（ $=9$ ）が、3 番目のブロックのデータ（D3-1 ~ D3-12）を入力する期間におけるステップ幅として、アドレスカウンタ 3 へ出力されることとなる。

【0072】このため、3 番目のブロックのデータ（D3-1 ~ D3-12）を入力する期間において、アドレスカウンタ 3 からは、0, 9, 7, 5, 3, 1, 10, 8, 6, 4, 2, 11 の順にアドレスが出力されることとなる。尚、この期間において、出力されるアドレスが 9 の次に 7 へと変化しているのは、9 に現在のステップ幅である 9 を加算した値（ $=18$ ）が、ブロックサイズ（ $=12$ ）以上となるため、この場合には、減算器 1 5 の演算結果であって、18 についてブロックサイズ - 1（ $=11$ ）のモジュロを取った値である 7 が、次のカウント

値としてF/F群11にラッチされるからである。そして、このことは、出力されるアドレスが7の次に5へと変化する場合や、5の次に3へと変化する場合等についても同様である。

【0073】また、この期間においては、ステップ幅演算器5のF/F群23、加算器25、減算器27、コンパレータ29、33、及びセクタ31、35からなる回路部分の作用により、アドレスカウンタ3からのアドレスがカラム数M(=3)よりも小さい0、1、2の各場合に、F/F群21から現在出力されているステップ幅(=9)が累積加算されると共に、その累積加算値についてブロックサイズ-1(=11)のモジュロを取った値(=5)が、次のブロックに対する新たなステップ幅としてF/F群23に記憶される。尚、この場合、F/F群23に記憶される値は、0→9→7→5と変化する。

【0074】そして、図4には示されていないが、その後、アドレスカウンタ3から出力されるアドレスが11から0に戻ると、そのタイミングで、F/F群23に記憶されている新たなステップ幅(=5)がF/F群21にラッチされると共に、F/F群23の出力値が0に戻り、F/F群21にラッチされた新たなステップ幅(=5)が、4番目のブロックのデータを入力する期間におけるステップ幅として、アドレスカウンタ3へ出力されることとなる。このため、4番目のブロックのデータを入力する期間において、アドレスカウンタ3からは、0、5、10、4、9、3、8、2、7、1、6、11の順にアドレスが出力されることとなる。

【0075】以上のように、本第1実施形態のインターリーブアドレス発生器1では、アドレスカウンタ3からクロックに同期して出力されるアドレスのステップ幅が、その出力されるアドレスがブロックサイズ-1(=N×M-1)から初期値0に戻る毎であって入力データの1ブロック分毎に、ステップ幅演算器5によって変更されることとなる。

【0076】よって、このインターリーブアドレス発生器1によれば、1番目のブロックのデータを入力する期間においては、アドレスカウンタ3から出力されるメモリの各アドレスに入力データを順次書き込み、2番目以降の各ブロックのデータを入力する期間においては、アドレスカウンタ3から出力されるメモリの各アドレスからデータを読み出すと共に、その読み出したアドレスに現在の入力データを上書きすることにより、図1を用いて既述した如く、メモリからのデータの読み出しが、そのデータを書き込んだ時とは異なるステップ幅でアドレスリングされることとなる。例えば、図4に示した例の場合において、1ブロック分の入力データに対し1番目のものから順にD1、D2、D3、D4、D5、D6、D7、D8、D9、D10、D11、D12というインデックスを付けるとすると、順次入力される各ブロックの入力

データは、メモリから読み出されることにより、夫々、D1、D4、D7、D10、D2、D5、D8、D11、D3、D6、D9、D12という順序に並べ替えられることとなる。

【0077】このため、本第1実施形態のインターリーブアドレス発生器1によれば、当該インターリーブアドレス発生器1とメモリとを1組備えるだけで、複数のブロックのデータに対する連続したインターリーブ処理が可能となり、インターリーブ処理を行うための回路(インターリーブ回路)を小型化することができる。

【0078】尚、本第1実施形態では、アドレスカウンタ3が、アドレスカウンタ手段に相当しており、そのアドレスカウンタ3の動作が、アドレスカウンタ処理に相当している。また、ステップ幅演算器5が、ステップ幅演算手段に相当しており、そのステップ幅演算器5の動作が、ステップ幅設定処理に相当している。そして、ステップ幅演算器5を構成する各部のうちで、F/F群23、加算器25、減算器27、コンパレータ29、33、セクタ31、35、及び論理回路41が、請求項2に記載のステップ幅設定手段に相当し、コンパレータ37が、請求項2に記載の検出手段に相当し、F/F群21及びセクタ39が、請求項2に記載のステップ幅出力手段に相当している。

【0079】ところで、上記第1実施形態のインターリーブアドレス発生器1では、アドレスカウンタ3へ現在出力しているステップ幅にカラム数Mを乗じた値を、現在のステップ幅をカラム数M回分だけ累積加算することによって求めたが、現在のステップ幅とカラム数Mとを直接乗算するようにしても良い。

【0080】そこで次に、現在のステップ幅にカラム数Mを乗じた値を、現在のステップ幅とカラム数Mとを乗算器にて乗算することにより求めるようにした第2実施形態のインターリーブアドレス発生器について、具体的に説明する。まず、図5は、本第2実施形態のインターリーブアドレス発生器43を表す回路図である。尚、図5において、前述した第1実施形態のインターリーブアドレス発生器1と同じ構成要素については、同一の符号を付しているため、詳細な説明は省略する。

【0081】図5に示すように、本第2実施形態のインターリーブアドレス発生器43は、第1実施形態のインターリーブアドレス発生器1と比較して、ステップ幅演算器45の構成だけが異なっており、アドレスカウンタ3は全く同じである。そして、本第2実施形態のステップ幅演算器45は、第1実施形態のステップ幅演算器5と比較して、加算器25、減算器27、コンパレータ29、33、セクタ31、35、及び論理回路41の代わりに、F/F群21の出力値(即ち、現在のステップ幅)とカラム数入力端子T3からのカラム数(=M)とを乗算する乗算器47と、アドレスカウンタ3のカウント値(即ち、アドレスカウンタ3のF/F群11から

メモリへ出力されているアドレス)が0である場合に、ハイレベルのセレクト信号を出力し、アドレスカウンタ3のカウント値が0でない場合に、ロウレベルのセレクト信号を出力するコンパレータ49と、コンパレータ49からのセレクト信号がハイレベルの場合に、乗算器47の演算結果を選択して出力し、コンパレータ49からのセレクト信号がロウレベルの場合に、F/F群23の出力を選択して出力するセクタ51と、そのセクタ51の出力から、ブロックサイズ入力端子T2より入力されるブロックサイズ-1 ($=N \times M - 1$) を減算する減算器53と、セクタ51の出力とブロックサイズ入力端子T2からのブロックサイズ-1とを大小比較して、セクタ51の出力がブロックサイズ-1 ($=N \times M - 1$) 以下である場合に、ハイレベルのセレクト信号を出力し、そうでなければ(つまり、セクタ51の出力がブロックサイズ ($=N \times M$) 以上である場合に)、ロウレベルのセレクト信号を出力するコンパレータ55と、コンパレータ55からのセレクト信号がハイレベルの場合に、セクタ51の出力をF/F群23のデータ端子(D)に入力させ、コンパレータ55からのセレクト信号がロウレベルの場合に、減算器53の演算結果をF/F群23のデータ端子(D)に入力させるセクタ57と、を備えている。

【0082】そして、このステップ幅演算器45では、F/F群23のリセット端子(RESET)に、リセット信号入力端子T4からのリセット信号が直接入力されるようになっている。このように構成されたステップ幅演算器45においては、リセット信号入力端子T4からハイレベルのリセット信号が入力されると、F/F群23では、その各フリップフロップのリセット端子(RESET)に入力される信号がハイレベルとなり、また、F/F群21では、1ビット目のフリップフロップのプリセット端子(SET)と、それ以外のビットのフリップフロップのリセット端子(RESET)とに夫々入力される信号がハイレベルとなる。

【0083】このため、第1実施形態のステップ幅演算器5と同様に、リセット信号入力端子T4からハイレベルのリセット信号が入力されると、クロック入力端子T1からのクロックに同期して、F/F群23の出力値は全ビットが0となる。また、F/F群21の出力値は、1ビット目だけが1で、それ以外のビットが0となり、この結果、アドレスカウンタ3へ出力されるステップ幅は初期値としての1になる(図6参照)。

【0084】そして、その後、リセット信号入力端子T4からのリセット信号がロウレベルになって、リセットが解除されると、F/F群23は、セクタ57の出力をクロックに同期してラッチし出力する。ここで、本第2実施形態のステップ幅演算器45においては、アドレスカウンタ3のカウント値が0である場合に、セクタ51から、乗算器47によってF/F群21の出力値と

カラム数 ($=M$) とが乗算された乗算結果が出力され、更に、その乗算器47の乗算結果がブロックサイズ-1 ($=N \times M - 1$) 以下であれば、その乗算結果がそのままセクタ57からF/F群23のデータ端子(D)へ出力され、逆に、乗算器47の演算結果がブロック数 ($=N \times M$) 以上であれば、減算器53によって乗算器47の乗算結果からブロックサイズ-1 ($=N \times M - 1$) が減算された演算結果が、セクタ57からF/F群23のデータ端子(D)へ出力される。

10 【0085】また、アドレスカウンタ3のカウント値が0でない場合には、セクタ51から、F/F群23の出力の方が選択されて出力され、更に、そのF/F群23の出力値がブロックサイズ-1 ($=N \times M - 1$) 以下であれば、そのF/F群23の出力がそのままセクタ57からF/F群23のデータ端子(D)へ出力され、逆に、F/F群23の出力値がブロック数 ($=N \times M$) 以上であれば、減算器53によってF/F群23の出力値からブロックサイズ-1 ($=N \times M - 1$) が減算された演算結果が、セクタ57からF/F群23のデータ端子(D)へ出力される。

20 【0086】このため、F/F群23は、乗算器47、減算器53、コンパレータ49、55、及びセクタ51、57と共同して動作することにより、F/F群21から出力されているステップ幅とカラム数Mとを乗算した乗算値についてブロックサイズ-1 ($=N \times M - 1$) のモジュロを取った値を、次のブロックに対するインターリーブ処理時の新たなステップ幅として記憶することとなる。そして、この動作により、アドレスカウンタ3へ現在出力しているステップ幅にカラム数 ($=M$) を乗じた値についてブロックサイズ-1 ($=N \times M - 1$) のモジュロを取った値が、次のブロックに対するインターリーブ処理時の新たなステップ幅として設定される。

30 【0087】一方、当該ステップ幅演算器45においても、第1実施形態のステップ幅演算器5と同様に、リセット信号入力端子T4からのリセット信号がハイレベルからロウレベルになってリセットが解除されると、F/F群21は、コンパレータ37からのセレクト信号に応じて動作するセクタ39の出力を、クロックに同期してラッチし、ステップ幅として出力する。そして、セクタ39は、コンパレータ37からのセレクト信号がロウレベルの場合(即ち、アドレスカウンタ3のカウント値がブロックサイズ-1 ($=N \times M - 1$) でない場合)には、F/F群21の出力をF/F群21のデータ端子(D)に入力させ、また、コンパレータ37からのセレクト信号がハイレベルの場合(即ち、アドレスカウンタ3のカウント値がブロックサイズ-1 ($=N \times M - 1$) である場合)に、F/F群23の出力をF/F群21のデータ端子(D)に入力させる。

40 【0088】よって、本第2実施形態のステップ幅演算器45においても、F/F群21は、アドレスカウンタ

3のカウンタ値がブロックサイズ-1 ($=N \times M - 1$) になると、次のクロックタイミング (即ち、アドレスカウンタ3のカウンタ値が初期値0に戻るタイミング) で、F/F群23に記憶されている次のブロックに対する新たなステップ幅をラッチし、そのラッチしたステップ幅を、アドレスカウンタ3のカウンタ値が次にブロックサイズ-1 ($=N \times M - 1$) から初期値0へ戻るまでの1ブロック分の期間中、アドレスカウンタ3へ出力することとなる。

【0089】ここで、図6は、本第2実施形態のインターリーブアドレス発生器43の動作を、行数 (ロー数) $N=4$ 及び列数 (カラム数) $M=3$ のブロックサイズでインターリーブ処理を行う場合を例に挙げて表すタイムチャートである。そして、図6に示すように、本第2実施形態のインターリーブアドレス発生器43では、F/F群23の出力値が、1番目のブロックのデータ (D1-1 ~ D1-12) を入力する期間の開始から2つ目のクロックタイミングにおいて、0から3へと変化し、2番目のブロックのデータ (D2-1 ~ D2-12) を入力する期間の開始から2つ目のクロックタイミングにおいて、3から9へと変化し、更に、3番目のブロックのデータ (D3-1 ~ D3-12) を入力する期間の開始から2つ目と3つ目の各クロックタイミングにおいて、9→16→5と順次変化する点のみが、第1実施形態のインターリーブアドレス発生器1と異なっている。

【0090】そして、このような本第2実施形態のインターリーブアドレス発生器43によっても、アドレスカウンタ3からは、第1実施形態のインターリーブアドレス発生器1と全く同様の順序でアドレスが出力されることとなり、当該インターリーブアドレス発生器43とメモリとを1組備えるだけで、複数のブロックのデータに対する連続したインターリーブ処理が可能となる。

【0091】尚、本第2実施形態では、アドレスカウンタ3が、アドレスカウント手段に相当しており、そのアドレスカウンタ3の動作が、アドレスカウント処理に相当している。また、ステップ幅演算器45が、ステップ幅演算手段に相当しており、そのステップ幅演算器45の動作が、ステップ幅設定処理に相当している。そして、ステップ幅演算器45を構成する各部のうちで、F/F群23、乗算器47、減算器53、コンパレータ49、55、及びセクタ51、57が、請求項3に記載のステップ幅設定手段に相当し、コンパレータ37が、請求項3に記載の検出手段に相当し、F/F群21及びセクタ39が、請求項3に記載のステップ幅出力手段に相当している。

【0092】以上、本発明の一実施形態について説明したが、本発明は、前述した実施形態に限定されるもので

はなく、種々の形態を採り得ることは言うまでもない。例えば、前述した各実施形態では、ステップ幅演算器5、45からアドレスカウンタ3へ出力されるステップ幅の初期値が1であったが、ステップ幅の初期値は、1以上の整数であれば良い。

【0093】また、インターリーブ処理に用いるメモリの先頭アドレスが0番地でない場合には、アドレスカウンタ3から出力されるアドレスに、メモリの先頭アドレスの値を加算し、その加算後のアドレスをメモリのアドレス入力端子に供給するようにすれば良い。

【図面の簡単な説明】

【図1】 本発明の作用を説明する説明図である。

【図2】 第1実施形態のインターリーブアドレス発生器の構成を表すブロック図である。

【図3】 第1実施形態のインターリーブアドレス発生器の構成を表す回路図である。

【図4】 図3のインターリーブアドレス発生器の動作を表すタイムチャートである。

【図5】 第2実施形態のインターリーブアドレス発生器の構成を表す回路図である。

【図6】 図5のインターリーブアドレス発生器の動作を表すタイムチャートである。

【図7】 従来のインターリーブアドレス発生器を用いたインターリーブ回路の基本構成を表すブロック図である。

【図8】 図7のインターリーブ回路の動作を説明する説明図である。

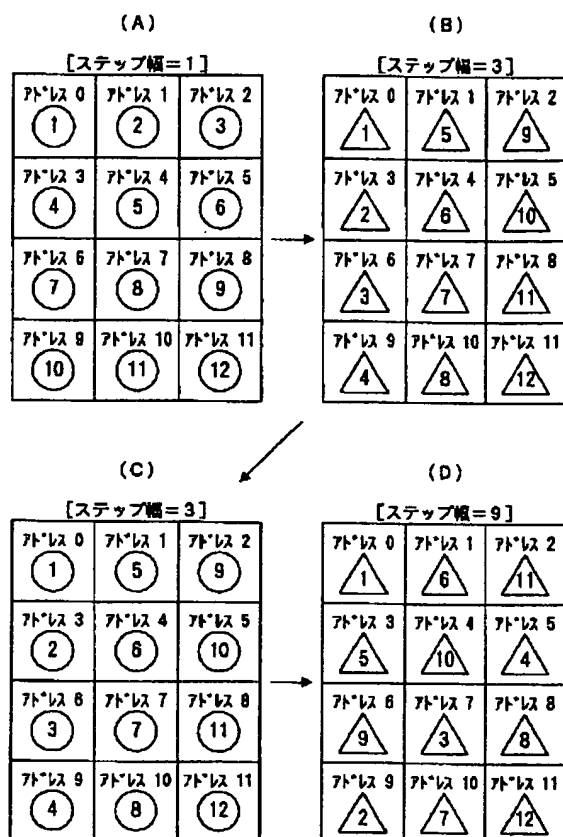
【図9】 複数のブロックのデータを連続してインターリーブ処理するための、従来のインターリーブ回路の構成を表すブロック図である。

【図10】 図9のインターリーブ回路の動作を表すタイムチャートである。

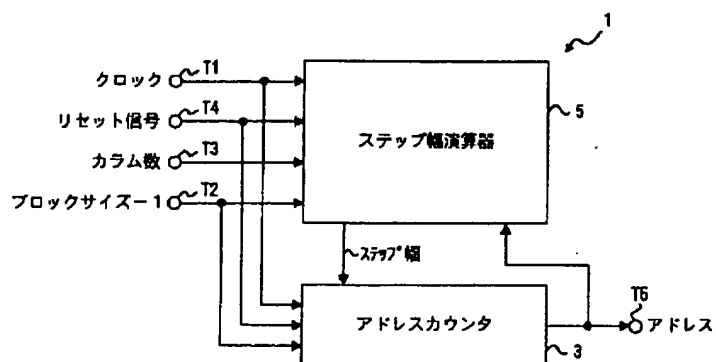
【符号の説明】

1, 43…インターリーブアドレス発生器
3…アドレスカウンタ 5, 45…ステップ幅演算器
11, 21, 23…フリップフロップ群 (F/F群)
13, 25…加算器 15, 27, 53…減算器
47…乗算器
17, 29, 33, 37, 49, 55…コンパレータ
19, 31, 35, 39, 51, 57…セクタ 4
1…論理和回路
T1…クロック入力端子 T2…ブロックサイズ入力端子
T3…カラム数入力端子 T4…リセット信号入力端子
T5…アドレス出力端子

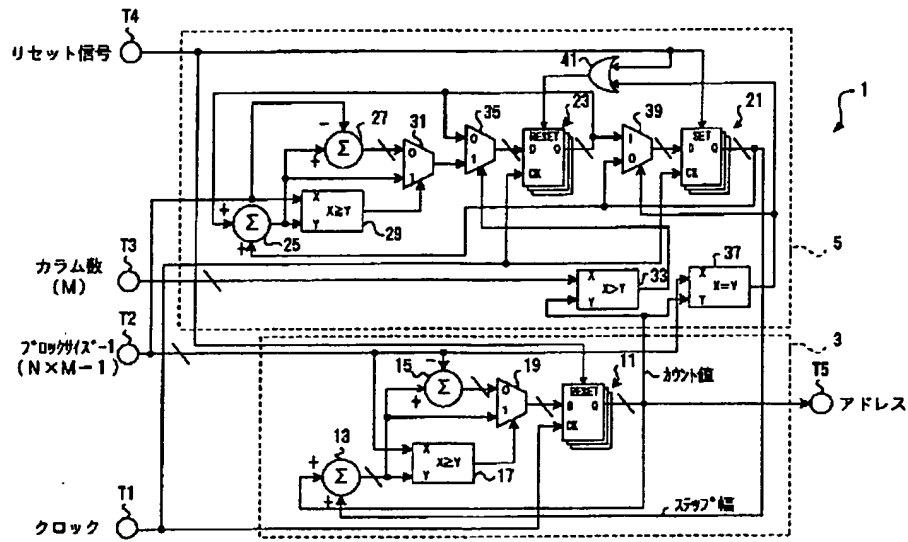
【図1】



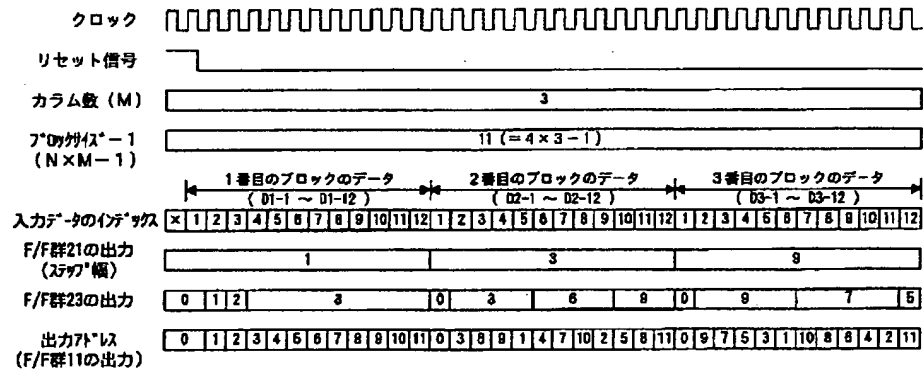
【図2】



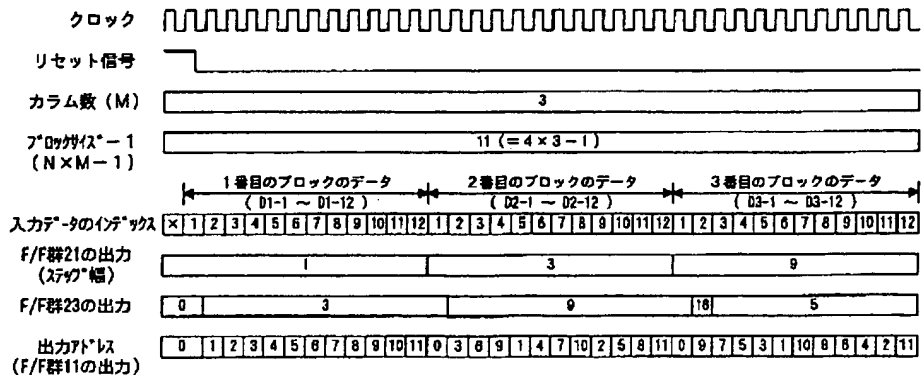
【図3】



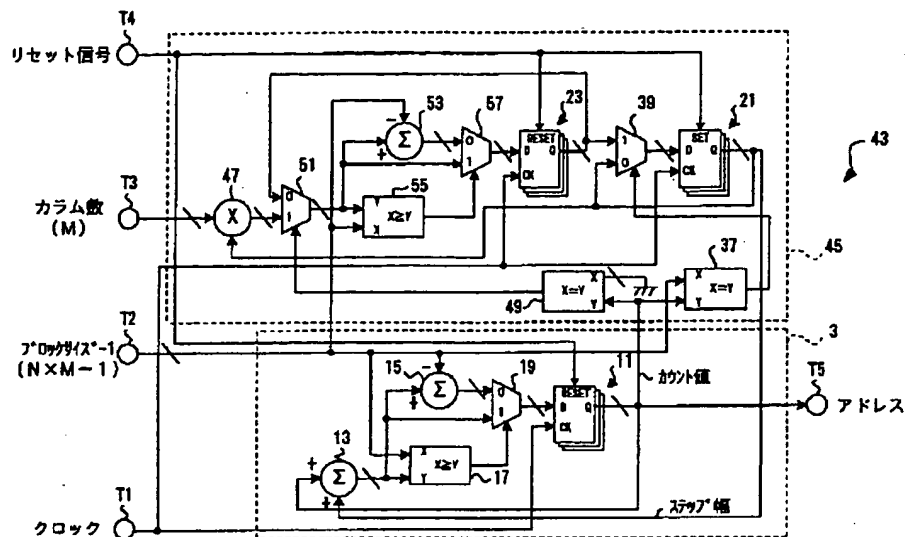
【図4】



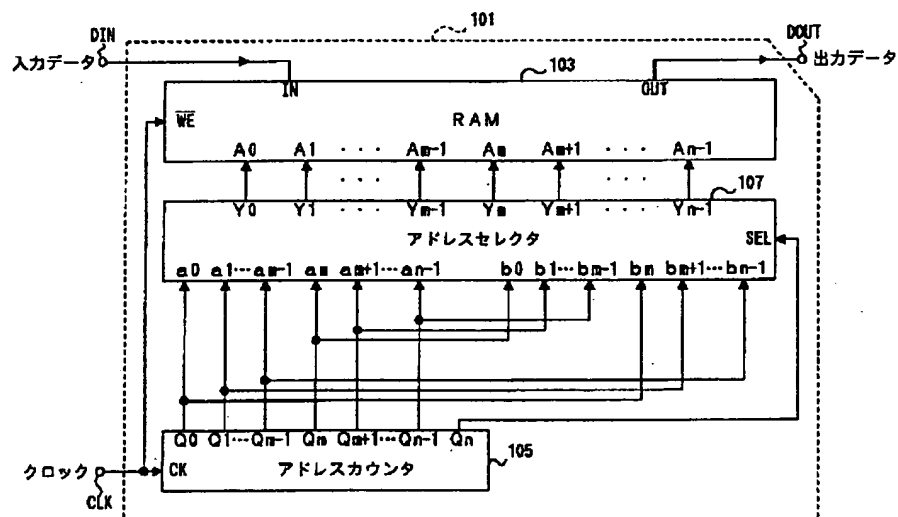
【図6】



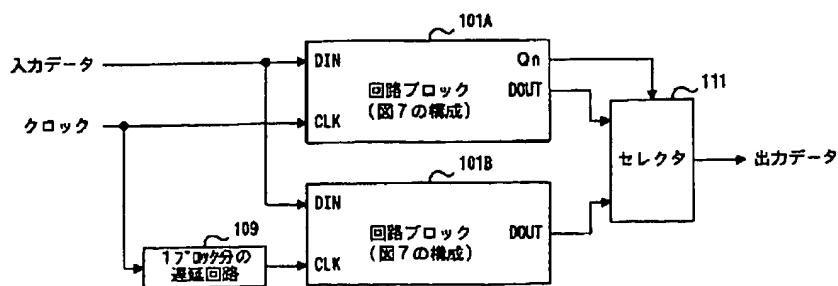
【図 5】



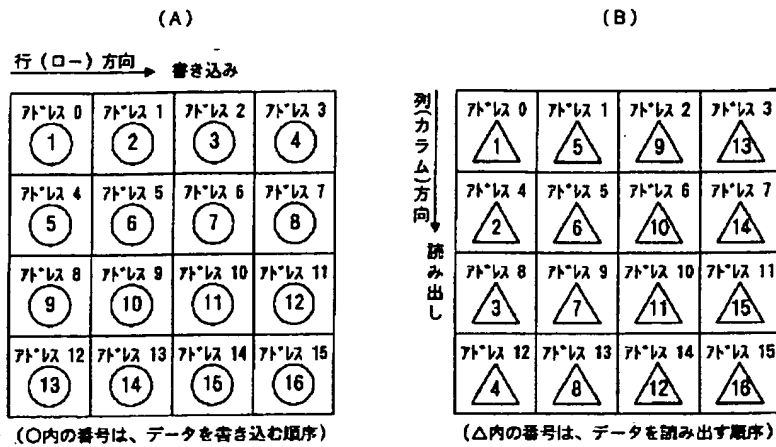
【図 7】



【図 9】



【図8】



【図10】

